# A UNIVERSAL OBJECT-ORIENTED MEDICAL DATABASE SYSTEM

V.A. Nagin, I.V. Potapov, S. V. Selishchev
Biomedical Systems Department, Moscow State Institute of Electronic Technology, Moscow, Russia

*Abstract*-**The paper presents and discusses a universal client/server object-oriented patient database system for clinical and personal applications. The developed system makes it possible to manage all patient specific data in the unified way by means of object-oriented representation of heterogeneous medical data, which include patient details, examination results and doctor visits history. The presented implementation includes a database server and a universal client application, which serves as a container for manifold medical applications that require database services. This approach enables to simplify drastically the implementation of database support in client applications and to put the main computational burden on the container application, which communicates with its clients by means of defined interface protocols.**

***Keywords* – database management, OLE, COM+, client/server, biomedical data, distributed database.**

## I. INTRODUCTION

During past several years the growth of costs of medical care system maintenance has led medical information system developers to the new approach based on patient-oriented architecture. The result of these tendencies is involving of a growing number of new organizations with their heterogeneous systems in the common informational environment. At present the design and development of Database Management Systems (DBMS) is one the most rapidly growing sphere of IT technologies in medical care [1].

Currently two approaches to implementation of complex medical DBMS solutions are available. First one is complete reequipment with installation of new diagnostic systems preferably produced and supported by the single manufacturer. It enables to use the integration possibilities offered by the single supplier more effectively. Second approach is slightly more complicated for the implementation because it implies the integration of each particular medical application with a common DBMS. Moreover, not every application may offer such integration features. In case of presence of medical diagnostic equipment that doesn't satisfy entirely or in part the modern standards of medical information interchange the second approach has no alternative.

The approach we developed makes it possible to solve this task minimizing the expenditures for the system integration and providing at the same time smooth transition for obsolete equipment to the new level of medical information system interoperability. The main principle of the named approach is that the medical application carrying out diagnostic functions is itself responsible for establishing and maintaining interchange with a databank, preserving and visualizing data in the required form. In case of absence of the required application on the particular workstation it can be installed automatically. Common functionality provided by a hosting operating system (OS) is shared between applications designed for this OS what provides extended flexibility for data interchange facility implementation. Currently the developed system makes it possible to integrate applications designed for MS Windows[1] operating system family and DOS focusing, however, on Windows applications. Usage of ActiveX documents architecture permits to embed medical application data into a database immediately and provides basic visual representation [2]. Visual representation is platform-independent and doesn't require any specific visualization software.

The presented system was designed to satisfy following requirements.

Functional requirements:

1) Minimal demands for hosted medical applications;
2) Easy to learn user interface;
3) Wide application sphere (clinical, laboratory and scientific use).

Architectural requirements:

1) Modular structure;
2) Internet-aware design;
3) High security level.

As a result of these requirements following system configuration was created.

## II. SYSTEM STRUCTURE

System consists of:

1) An enterprise server:
   a. SQL server based enterprise database;
   b. Applications and user management database;
   c. Automated database analysis and statistics wizards set;
   d. Thin web browser based database client [3].
2) A universal database client application with a local patient and medical records database:
   a. Internal replication utility applicable to selected patient records in the enterprise database and newly created patient records [4];
   b. Tools for data interchange and synchronization (IrDA, e-mail);
   c. Authentication and data protection tools (Unified Windows smart cards interface is used).

The detailed system structure is presented on Fig. 1 below.

# Report Documentation Page

| Report Date | Report Type | Dates Covered (from... to) |
|---|---|---|
| 25 Oct 2001 | N/A | - |

| | |
|---|---|
| **Title and Subtitle**<br>A Universal Object-Oriented Medical Database System | **Contract Number** |
| | **Grant Number** |
| | **Program Element Number** |
| **Author(s)** | **Project Number** |
| | **Task Number** |
| | **Work Unit Number** |
| **Performing Organization Name(s) and Address(es)**<br>Biomedical Systems Department Moscow State Institute of Electronic Technology Moscow, Russia | **Performing Organization Report Number** |
| **Sponsoring/Monitoring Agency Name(s) and Address(es)**<br>US Army Research, Development & Standardization Group (UK) PSC 802 Box 15 FPO AE 09499-1500 | **Sponsor/Monitor's Acronym(s)** |
| | **Sponsor/Monitor's Report Number(s)** |

**Distribution/Availability Statement**
Approved for public release, distribution unlimited

**Supplementary Notes**
Papers from 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, October 25-28, 2001, held in Istanbul, Turkey. See also ADM001351 for entire conference on cd-rom., The original document contains color images.

**Abstract**

**Subject Terms**

| | |
|---|---|
| **Report Classification**<br>unclassified | **Classification of this page**<br>unclassified |
| **Classification of Abstract**<br>unclassified | **Limitation of Abstract**<br>UU |

**Number of Pages**
3

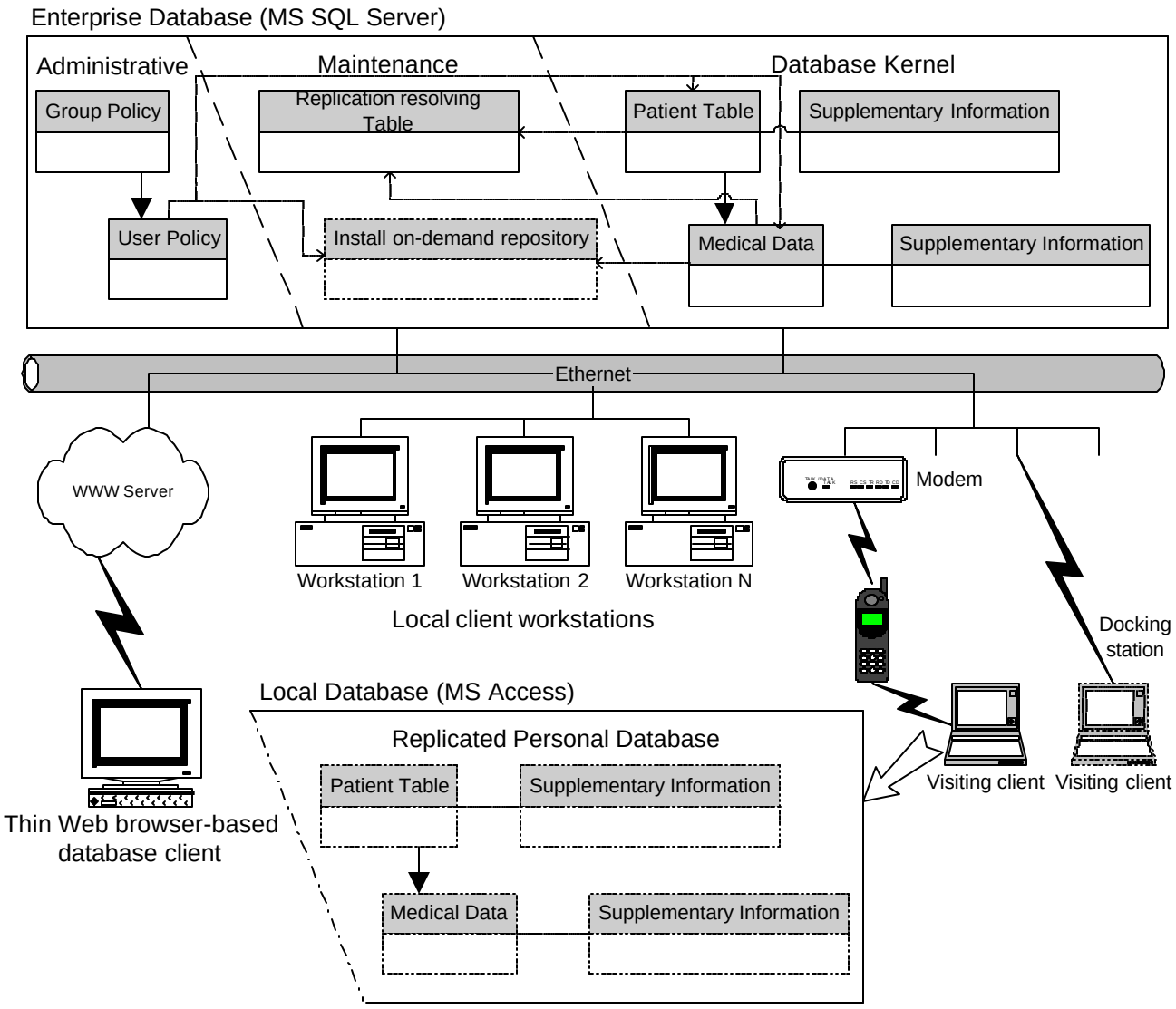


Fig. 1. Universal medical DBMS structure

Database structures of the enterprise server database and the client application database are identical to a great extent. It can be logically divided into three main sub storages: medical stuff details with access rights and keys; patients list with all linked medical records; additional storage for reviewing software and operation log files. All medical records are stored as OLE embedded documents, for applications that do not support this technique highly customizable ActiveX document wrapper is provided. An internally replicated database is used to store patient and medical information, which can be used by doctors out of a clinic.

Enterprise server performs all transactional and replication operations on database, holds user access rights and is used for authentication. Standard Windows smart card authentication procedure is used to achieve top level of security.

## III. CONCLUSION

Implementation of the offered approach permitted to integrate a number of medical applications in one informational medium. It is worthy of note that this solution isn't deprived of disadvantages. In particular, multi-platform integration (i.e. support of different UNIX systems) is complicated or even nearly impossible. Luckily, a great majority of applications based on these operating systems supports a commonly accepted interchange interfaces such as HL7 and DICOM. Therefore these protocols can be used to implement interoperability features.

1 Microsoft Corporation, Redmond WA

REFERENCES

[1] Hurson AR, Pakzad SH, Cheng J.-b. Object-oriented database management systems: Evolution and performance issues. IEEE Computer 1993; 26(2):

[2] Distributed Component Object Model Protocol Version 1.0 Microsoft, January 1998.

[3] George R. Thoma, A Client/Server System for Internet Access to Biomedical Text/Image Databanks, Computerized Medical Imaging and Graphics, Vol.20 No.4, pp.259-268, 1996.

[4] J. Duchene, MicroPACS: a PC-based small PACS implementation. Med-Biol-Eng-Comput. 31(3) 268-76 (1993)